**HTML** stands for **Hyper Text Markup Language**, which is the most widely used language on Web to develop web pages. **HTML** was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

**Why to Learn HTML?**

Originally, **HTML** was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers. Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

**HTML** is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning HTML:

* **Create Web site**- You can create a website or customize an existing web template if you know HTML well.
* **Become a web designer**- If you want to start a carrer as a professional web designer, HTML and CSS designing is a must skill.
* **Understand web**- If you want to optimize your website, to boost its speed and performance, it is good to know HTML to yield best results.
* **Learn other languages**- Once you understands the basic of HTML then other related technologies like javascript, php, or angular are become easier to understand.

**Hello World using HTML.**

Just to give you a little excitement about HTML, I'm going to give you a small conventional **HTML Hello T.I.P** program.

<!DOCTYPE html>

<html>

   <head>

      <title>This is document title</title>

   </head>

   <body>

      <h1>This is a heading</h1>

      <p>Hello T.I.P!</p>

   </body>

</html>

**Applications of HTML**

As mentioned before, HTML is one of the most widely used language over the web. I'm going to list few of them here:

* **Web pages development**- HTML is used to create pages which are rendered over the web. Almost every page of web is having html tags in it to render its details in browser.
* **Internet Navigation**- HTML provides tags which are used to navigate from one page to another and is heavily used in internet navigation.
* **Responsive UI**- HTML pages now-a-days works well on all platform, mobile, tabs, desktop or laptops owing to responsive design strategy.
* **Offline support**HTML pages once loaded can be made available offline on the machine without any need of internet.
* **Game development**- HTML5 has native support for rich experience and is now useful in gaming development arena as well.

**Basic of HTML**

HTML stands for **H**yper**t**ext **M**arkup **L**anguage, and it is the most widely used language to write Web Pages.

* **Hypertext**refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext.
* As its name suggests, HTML is a **Markup Language**which means you use HTML to simply "mark-up" a text document with tags that tell a Web browser how to structure it to display.

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers.

Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

**Basic HTML Document**

In its simplest form, following is an example of an HTML document −

<!DOCTYPE html>

<html>

   <head>

      <title>This is document title</title>

   </head>

    <body>

      <h1>This is a heading</h1>

      <p>Document content goes here.....</p>

   </body>

      </html>

**HTML Tags**

As told earlier, HTML is a markup language and makes use of various tags to format the content. These tags are enclosed within angle braces **<Tag Name>**. Except few tags, most of the tags have their corresponding closing tags. For example, **<html>** has its closing tag **</html>** and **<body>** tag has its closing tag **</body>** tag etc.

Above example of HTML document uses the following tags −

|  |  |
| --- | --- |
| **Sr.No** | **Tag & Description** |
| 1 | **<!DOCTYPE...>**  This tag defines the document type and HTML version. |
| 2 | **<html>**  This tag encloses the complete HTML document and mainly comprises of document header which is represented by <head>...</head> and document body which is represented by <body>...</body> tags. |
| 3 | **<head>**  This tag represents the document's header which can keep other HTML tags like <title>, <link> etc. |
| 4 | **<title>**  The <title> tag is used inside the <head> tag to mention the document title. |
| 5 | **<body>**  This tag represents the document's body which keeps other HTML tags like <h1>, <div>, <p> etc. |
| 6 | **<h1>**  This tag represents the heading. |
| 7 | **<p>**  This tag represents a paragraph. |

To learn HTML, you will need to study various tags and understand how they behave, while formatting a textual document. Learning HTML is simple as users have to learn the usage of different tags in order to format the text or images to make a beautiful webpage.

World Wide Web Consortium (W3C) recommends to use lowercase tags starting from HTML 4.

 HTML Document Structure

A typical HTML document will have the following structure −

<html>

     <head>

      Document header related tags

   </head>

     <body>

      Document body related tags

   </body>

  </html>

The <!DOCTYPE> Declaration

The <!DOCTYPE> declaration tag is used by the web browser to understand the version of the HTML used in the document. Current version of HTML is 5 and it makes use of the following declaration −

<!DOCTYPE html>

There are many other declaration types which can be used in HTML document depending on what version of HTML is being used. We will see more details on this while discussing <!DOCTYPE...> tag along with other HTML tags.

**Heading Tags**

Any document starts with a heading. You can use different sizes for your headings. HTML also has six levels of headings, which use the elements **<h1>, <h2>, <h3>, <h4>, <h5>,** and **<h6>**. While displaying any heading, browser adds one line before and one line after that heading.

**Example**

<!DOCTYPE html><html>

<head>

<title>Heading Example</title>

</head>

<body>

<h1>This is heading 1</h1>

<h2>This is heading 2</h2>

<h3>This is heading 3</h3>

<h4>This is heading 4</h4>

<h5>This is heading 5</h5>

<h6>This is heading 6</h6>

</body>

</html>

## Basic HTML Document

In its simplest form, following is an example of an HTML document −

<!DOCTYPE html><html>    <head>      <title>This is document title</title>   </head>          <body>      <h1>This is a heading</h1>      <p>Document content goes here.....</p>   </body>       </html>

# **This is a heading**

Document content goes here.....

## ****HTML Tags****

As told earlier, HTML is a markup language and makes use of various tags to format the content. These tags are enclosed within angle braces **<Tag Name>**. Except few tags, most of the tags have their corresponding closing tags. For example, **<html>** has its closing tag **</html>** and **<body>** tag has its closing tag **</body>** tag etc.

Above example of HTML document uses the following tags −

|  |  |
| --- | --- |
| **Sr.No** | **Tag & Description** |
| 1 | **<!DOCTYPE...>**  This tag defines the document type and HTML version. |
| 2 | **<html>**  This tag encloses the complete HTML document and mainly comprises of document header which is represented by <head>...</head> and document body which is represented by <body>...</body> tags. |
| 3 | **<head>**  This tag represents the document's header which can keep other HTML tags like <title>, <link> etc. |
| 4 | **<title>**  The <title> tag is used inside the <head> tag to mention the document title. |
| 5 | **<body>**  This tag represents the document's body which keeps other HTML tags like <h1>, <div>, <p> etc. |
| 6 | **<h1>**  This tag represents the heading. |
| 7 | **<p>**  This tag represents a paragraph. |

To learn HTML, you will need to study various tags and understand how they behave, while formatting a textual document. Learning HTML is simple as users have to learn the usage of different tags in order to format the text or images to make a beautiful webpage.

World Wide Web Consortium (W3C) recommends to use lowercase tags starting from HTML 4.

**HTML Document Structure**

A typical HTML document will have the following structure −

<html>

     <head>

      Document header related tags

   </head>

     <body>

      Document body related tags

   </body>

  </html>

We will study all the header and body tags in subsequent chapters, but for now let's see what is document declaration tag.

The <!DOCTYPE> Declaration

The <!DOCTYPE> declaration tag is used by the web browser to understand the version of the HTML used in the document. Current version of HTML is 5 and it makes use of the following declaration −

<!DOCTYPE html>

There are many other declaration types which can be used in HTML document depending on what version of HTML is being used. We will see more details on this while discussing <!DOCTYPE...> tag along with other HTML tags.

# **HTML - Basic Tags**

## Heading Tags

Any document starts with a heading. You can use different sizes for your headings. HTML also has six levels of headings, which use the elements **<h1>, <h2>, <h3>, <h4>, <h5>,** and **<h6>**. While displaying any heading, browser adds one line before and one line after that heading.

### ****Example****

<!DOCTYPE html>

<html>

<head>

<title> Heading Example</title>

</head>

<body>

<h1>This is heading 1</h1>

<h2>This is heading 2</h2>

<h3>This is heading 3</h3>

<h4>This is heading 4</h4>

<h5>This is heading 5</h5>

<h6>This is heading 6</h6>

</body>

</html>

**This will produce the following result –**

# **This is heading 1**

## This is heading 2

### This is heading 3

#### **This is heading 4**

##### **This is heading 5**

###### **This is heading 6**

## ****Paragraph Tag****

The **<p>** tag offers a way to structure your text into different paragraphs. Each paragraph of text should go in between an opening <p> and a closing </p> tag as shown below in the example −

### ****Example****

<!DOCTYPE html><html>    <head>      <title>Paragraph Example</title>   </head>          <body>      <p>Here is a first paragraph of text.</p>      <p>Here is a second paragraph of text.</p>      <p>Here is a third paragraph of text.</p>   </body>       </html>

This will produce the following result −

Here is a first paragraph of text.

Here is a second paragraph of text.

Here is a third paragraph of text.

## Line Break Tag

Whenever you use the **<br />** element, anything following it starts from the next line. This tag is an example of an **empty** element, where you do not need opening and closing tags, as there is nothing to go in between them.

The <br /> tag has a space between the characters **br** and the forward slash. If you omit this space, older browsers will have trouble rendering the line break, while if you miss the forward slash character and just use <br> it is not valid in XHTML.

### ****Example****

<!DOCTYPE html>

<html>

<head>

<title>Line Break  Example</title>

</head>

  <body>

<p>Hello<br />         You delivered your assignment on-time.<br />         Thanks<br />         Cel Catch</p>

</body>

</html>

**This will produce the following result –**

Hello  
You delivered your assignment on time.  
Thanks  
Cel Catch

## ****Centering Content****

You can use **<center>** tag to put any content in the center of the page or any table cell.

### ****Example****

<!DOCTYPE html>

<html>

<head>

<title>Centring Content Example</title>

</head>

<body>

<p>This text is not in the center.</p>

<center>         <p>This text is in the center.</p>      </center>

</body>

</html>

**This will produce following result −**

This text is not in the center.

This text is in the center.

## ****Horizontal Lines****

Horizontal lines are used to visually break-up sections of a document. The **<hr>** tag creates a line from the current position in the document to the right margin and breaks the line accordingly.

For example, you may want to give a line between two paragraphs as in the given example below −

### ****Example****

<!DOCTYPE html>

<html>

<head>

<title>Horizontal Line Example</title>

</head>

<body>

<p>This is paragraph one and should be on top</p>

  <hr />

p>This is paragraph two and should be at bottom</p>

</body>

</html>

**This will produce the following result −**

This is paragraph one and should be on top

This is paragraph two and should be at bottom

Again **<hr />** tag is an example of the **empty** element, where you do not need opening and closing tags, as there is nothing to go in between them.

The **<hr />** element has a space between the characters **hr** and the forward slash. If you omit this space, older browsers will have trouble rendering the horizontal line, while if you miss the forward slash character and just use **<hr>** it is not valid in XHTML

## ****Preserve Formatting****

Sometimes, you want your text to follow the exact format of how it is written in the HTML document. In these cases, you can use the preformatted tag **<pre>**.

Any text between the opening **<pre>** tag and the closing **</pre>** tag will preserve the formatting of the source document.

### ****Example****

<!DOCTYPE html>

<html>

<head>

<title>Preserve Formatting Example</title>

</head>

<body>

<pre>         function testFunction( strText ){            alert (strText)         }      </pre>

</body>

</html>

**This will produce the following result −**

**function testFunction( strText ){    alert (strText) }**

Try using the same code without keeping it inside **<pre>...</pre>** tags

## ****Nonbreaking Spaces****

Suppose you want to use the phrase "12 Angry Men." Here, you would not want a browser to split the "12, Angry" and "Men" across two lines −

**An example of this technique appears in the movie "12 Angry Men."**

In cases, where you do not want the client browser to break text, you should use a nonbreaking space entity **&nbsp;** instead of a normal space. For example, when coding the "12 Angry Men" in a paragraph, you should use something similar to the following code −

### ****Example****

<!DOCTYPE html>

<html>

<head>

<title>Nonbreaking Spaces Example</title>

</head>

<body>      <p>An example of this technique appears in the movie "12&nbsp;Angry&nbsp; Men."</p>

</body>

</html>

**This will produce the following result −**

An example of this technique appears in the movie "12 Angry Men."

# **HTML - Elements**

An **HTML element** is defined by a starting tag. If the element contains other content, it ends with a closing tag, where the element name is preceded by a forward slash as shown below with few tags −

|  |  |  |
| --- | --- | --- |
| **Start Tag** | **Content** | **End Tag** |
| <p> | This is paragraph content. | </p> |
| <h1> | This is heading content. | </h1> |
| <div> | This is division content. | </div> |
| <br /> |  |  |

So here **<p>....</p>** is an HTML element, **<h1>...</h1>** is another HTML element. There are some HTML elements which don't need to be closed, such as **<img.../>**, **<hr />** and **<br />** elements. These are known as **void elements**.

HTML documents consists of a tree of these elements and they specify how HTML documents should be built, and what kind of content should be placed in what part of an HTML document.

## HTML Tag vs. Element

An HTML element is defined by a starting tag. If the element contains other content, it ends with a closing tag.

For example, **<p>** is starting tag of a paragraph and **</p>** is closing tag of the same paragraph but **<p>This is paragraph</p>** is a paragraph element.

## Nested HTML Elements

It is very much allowed to keep one HTML element inside another HTML element −

### ****Example****

<!DOCTYPE html>

<html>

<head>

<title>Nested Elements Example</title>   </head>

<body>      <h1>This is <i>italic</i> heading</h1>

<p>This is <u>underlined</u> paragraph</p>

</body>

</html>

**This will display the following result −**

# **This is**italic**heading**

This is underlined paragraph

**HTML** stands for **Hyper Text Markup Language**, which is the most widely used language on Web to develop web pages. **HTML** was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

## ****HTML - Attributes****

We have seen few HTML tags and their usage like heading tags **<h1>, <h2>,** paragraph tag **<p>** and other tags. We used them so far in their simplest form, but most of the HTML tags can also have attributes, which are extra bits of information.

An attribute is used to define the characteristics of an HTML element and is placed inside the element's opening tag. All attributes are made up of two parts − a **name** and a **value**

* The **name**is the property you want to set. For example, the paragraph **<p>** element in the example carries an attribute whose name is **align**, which you can use to indicate the alignment of paragraph on the page.
* The **value**is what you want the value of the property to be set and always put within quotations. The below example shows three possible values of align attribute: **left, center** and **right**.

Attribute names and attribute values are case-insensitive. However, the World Wide Web Consortium (W3C) recommends lowercase attributes/attribute values in their HTML 4 recommendation.

### ****Example****

<!DOCTYPE html>

<html>

 <head>

<title>Align Attribute  Example</title>

</head>

<body>

<p align = "left">This is left aligned</p>

<p align = "center">This is center aligned</p>

 <p align = "right">This is right aligned</p>

</body>

   </html>

**This will display the following result −**

This is left aligned

This is center aligned

This is right aligned

## Core Attributes

The four core attributes that can be used on the majority of HTML elements (although not all) are −

* Id
* Title
* Class
* Style

### ****The Id Attribute****

The **id** attribute of an HTML tag can be used to uniquely identify any element within an HTML page. There are two primary reasons that you might want to use an id attribute on an element −

* If an element carries an id attribute as a unique identifier, it is possible to identify just that element and its content.
* If you have two elements of the same name within a Web page (or style sheet), you can use the id attribute to distinguish between elements that have the same name.

 For now, let's use the id attribute to distinguish between two paragraph elements as shown below.

**Example**

## <p id = "html">This para explains what is HTML</p>

## <p id = "css">This para explains what is Cascading Style Sheet</p>

### ****The title Attribute****

The **title** attribute gives a suggested title for the element. They syntax for the **title** attribute is similar as explained for **id** attribute −

The behavior of this attribute will depend upon the element that carries it, although it is often displayed as a tooltip when cursor comes over the element or while the element is loading.

**Example**

<!DOCTYPE html>

<html>

<head>

<title>The title Attribute Example</title>   </head>

<body>

<h3 title = "Hello HTML!">Titled Heading Tag Example</h3>

</body>

</html>

**This will produce the following result −**

### ****Titled Heading Tag Example****

Now try to bring your cursor over "Titled Heading Tag Example" and you will see that whatever title you used in your code is coming out as a tooltip of the cursor. (Try this in a browser)

## ****The class Attribute****

The **class** attribute is used to associate an element with a style sheet, and specifies the class of element. You will learn more about the use of the class attribute when you will learn Cascading Style Sheet (CSS). So for now you can avoid it.

The value of the attribute may also be a space-separated list of class names.

For example −

## ****class = "className1 className2 className3"****

### ****The style Attribute****

The style attribute allows you to specify Cascading Style Sheet (CSS) rules within the element.

<!DOCTYPE html><html>    <head>      <title>The style Attribute</title>   </head>          <body>      <p style = "font-family:arial; color:#FF0000;">Some text...</p>   </body>       </html>

This will produce the following result −

Some text...

At this point of time, we are not learning CSS, so just let's proceed without bothering much about CSS. Here, you need to understand what are HTML attributes and how they can be used while formatting content.

## ****Internationalization Attributes****

There are three internationalization attributes, which are available for most (although not all) XHTML elements.

* dir
* lang
* xml:lang

## ****The dir Attribute****

The **dir** attribute allows you to indicate to the browser about the direction in which the text should flow. The dir attribute can take one of two values, as you can see in the table that follows −

|  |  |
| --- | --- |
| **Value** | **Meaning** |
| ltr | Left to right (the default value) |
| rtl | Right to left (for languages such as Hebrew or Arabic that are read right to left) |

**Example**

<!DOCTYPE html>

<html dir = "rtl">

<head>

<title>Display Directions</title>

</head>

<body>      This is how IE 5 renders right-to-left directed text.   </body>

</html>

**This will produce the following result −**

.This is how IE 5 renders right-to-left directed text

When dir attribute is used within the <html> tag, it determines how text will be presented within the entire document. When used within another tag, it controls the text's direction for just the content of that tag.

## ****The lang Attribute****

The **lang** attribute allows you to indicate the main language used in a document, but this attribute was kept in HTML only for backwards compatibility with earlier versions of HTML. This attribute has been replaced by the **xml:lang** attribute in new XHTML documents.

The values of the lang attribute are ISO-639 standard two-character language codes. Check [**HTML Language Codes: ISO 639** (Links to an external site.)](https://www.tutorialspoint.com/html/language_iso_codes.htm) for a complete list of language codes.

**Example**

[Live Demo (Links to an external site.)](http://tpcg.io/Hf0XPh)

<!DOCTYPE html><html lang = "en">    <head>      <title>English Language Page</title>   </head>    <body>      This page is using English Language   </body> </html>

This will produce the following result −

This page is using English Language

## ****The xml:lang Attribute****

The xml:lang attribute is the XHTML replacement for the lang attribute. The value of the xml:lang attribute should be an ISO-639 country code as mentioned in previous section.

## ****Generic Attributes****

Here's a table of some other attributes that are readily usable with many of the HTML tags.

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Options** | **Function** |
| align | right, left, center | Horizontally aligns tags |
| valign | top, middle, bottom | Vertically aligns tags within an HTML element. |
| bgcolor | numeric, hexidecimal, RGB values | Places a background color behind an element |
| background | URL | Places a background image behind an element |
| id | User Defined | Names an element for use with Cascading Style Sheets. |
| class | User Defined | Classifies an element for use with Cascading Style Sheets. |
| width | Numeric Value | Specifies the width of tables, images, or table cells. |
| height | Numeric Value | Specifies the height of tables, images, or table cells. |
| title | User Defined | "Pop-up" title of the elements. |

We will see related examples as we will proceed to study other HTML tags. For a complete list of HTML Tags and related attributes please check reference to [HTML Tags List (Links to an external site.)](https://www.tutorialspoint.com/html/html_tags_reference.htm).